Green Security Games Along Trails.

Author: Nicolás Betancourt
Supervisor: Prof. Mauricio Velasco

Proyecto de grado para obtener el título de Mágister en Matemáticas

Departmento de Matemáticas
Facultad de Ciencias
Universidad de los Andes

January 2, 2023

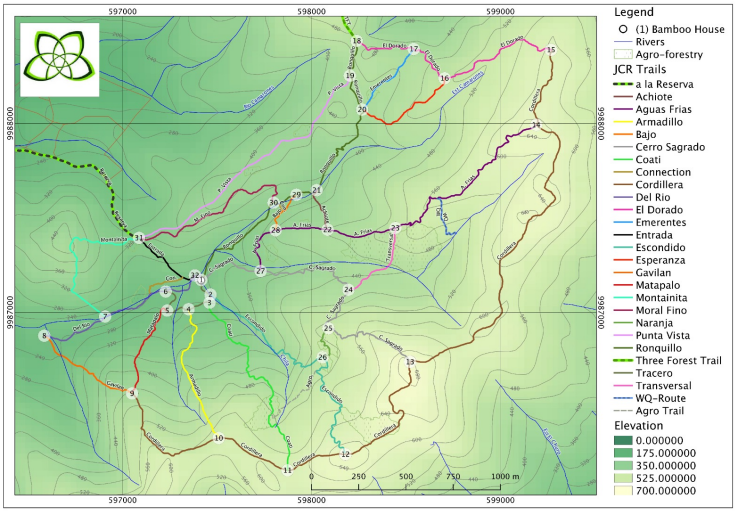
Contents

0 Acknowledgments

I am very grateful to Third Millenium Alliance, specially Ryan Lynch and Shawn McCracken, for providing data such as the files of the graph of trails of Jama Coaque and spreadsheets describing illegal activities on critical trails. They also furnished the time and space for me to hold interviews with the staff of the reserve in order to better understand the problem. We are also grateful to Jama Coaque's Operation Team comprised by Moises Tenorio, Dany Murillo, Sixto Lopez and Edilberto Marquez. They described to us their daily tasks as defenders of the reserve. Their insights and advice have been essential for this project. Finally, I wanted to thank Mauricio Velasco whose endless enthusiasm and support were essential for undertaking this project.

1 Introduction

Graph of trails of the Jamacoaque Ecological Reserve in Ecuador.



Mitigating the negative impact of the current climate crisis and biodiversity loss is not only one of the most compelling needs of recent years but also an unprecedented challenge due to its political, economic and technical dimensions. An essential resource against climate change which is also helpful in the preservation of earth's biodiversity is keeping large natural areas protected and intact for they provide shelter to a large variety of species and serve as carbon sinks and water reservoirs. Unfortunately, protected natural areas around the world are being constantly threatened by poaching, illegal logging, mining and wildlife trade. The extension of protected areas around the world is getting closer to the goal for 2020 set by the Convention on Biological Diversity, but the manpower allocated to monitor and safeguard these spaces are often insufficient [1]. This poses a challenge to those in charge of taking care of those areas because they have to optimally allocate the patrolling resources in extensive tracts of land and are often in great disadvantage against the attackers.

Computer scientists have made important contributions to such kinds of problems in the research field called Green Security Games. The main objective of this area is the design of decision rules for rangers in protected areas [2] so the surveillance is performed in the most critical places. Although there are many techniques for solving such problems, the main by-product of these algorithms is the construction of sequential patrolling routes within the area of the reserve.

A lot of work in this area has focused on open spaces like the African Savannah and

requires a discretization of the protected area. These approaches do not capture the reality of South American parks where, due to the density of the vegetation and the ruggedness of the terrain, traveling (by both rangers and poachers) is done only over a limited collection of available trails (a graph). Advised by a NGO devoted to the conservation of the last remaining acres of what once was the immense Ecuadorian Choco, we developed an adaptation of the ideas in [2] and in [3] for advising the rangers of the Jamacoaque Ecological Reserve on the routes that they should follow daily. Our methods [4] combine ideas from online learning and combinatorial optimization to learn from experience and adapt these routes to increase the frequency at which rangers thwart environmental crimes.

1.1 The ranger's problem

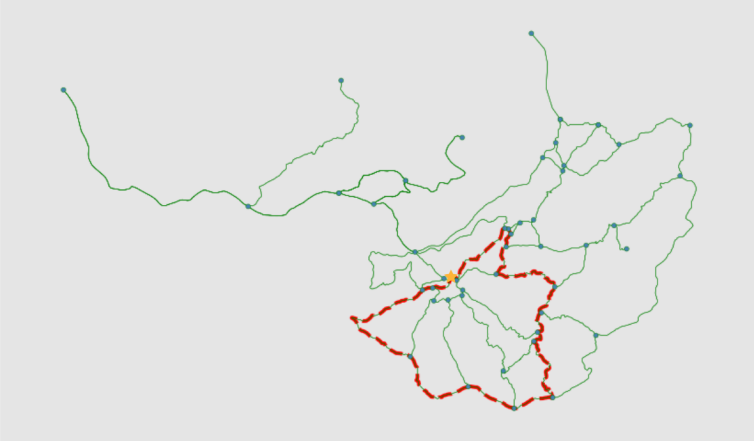
Jama Coaque Ecological Reserve is a growing protected area of more than 2,000 acres of threatened rainforest on the Pacific coast of Ecuador. Within the bast terrain of rugged topography there is a single spot at which people can find shelter known as The Bamboo House.

The bamboo house looks tiny in contrast with the immensity of the jungle.

On a daily basis, every available ranger must visit some of the trails looking for illegal activity. Every spot at the reserve other than the Bamboo House is unsuitable for resting

and the entrance trail of the reserve is only connected with the Bamboo house. This forces every ranger's patrolling route to start at the Bamboo House and eventually return there. Thus, the set of feasible patrolling routes for the staff of Jama Coaque are paths on the graph of trails that are cycle-shaped and traverses the Bamboo House .

A feasible route starting and ending at the Bamboo House.



More specifically, let G be the graph of trails and $C_G \in \mathbb{R}^{n\times M}$ (n = #trails, M = #circuits) the matrix of cycles of the graph of trails G i.e. the matrix such that its $j-$th column, $C^j \in \{0,1\}^n$ is the indicator vector of the $j-$th circuit

$$C_i^j = \begin{cases} 1 & \text{circuit } j \text{ traverses the } i\text{th trail} \\ 0 & \text{Otherwise} \end{cases}$$

under this notation, the ranger's problem is to choose columns of C_G sequentially so that the success of the patrolling policy is maximized and considering patrolling constraints. The way in which this success is measured will change for each of the alternatives that will be explored in this document. In all of them, however, we basically count the number of trails in the intersection of the routes of both agents, the poacher and the ranger and implement a penalization for resources wrongly allocated.

So far, the process of selecting circuits crossing the Bamboo House, has been executed manually in Jama Coaque. The staff list very few circuit-like paths routes and decides which route is to be followed based on experience. This is disadvantageous because it is nearly impossible to list all the feasible routes without computational aid (there are hundreds of thousands of them) and some routes that are frequently visited by invaders might be ignored accidentally in this process.

To close this section, let us highlight that this particular statement of The Ranger's Problem is suited for the specific needs of Jama Coaque's staff but we are certain that it

serves more general contexts for a wide variety of protected areas in Colombia, Ecuador and, more generally, South America.

2 Stackelberg security games.

The first method we will explore towards the solution of the ranger's problem is based on a green security game. The idea is to fix a probability distribution on the set of patrolling agendas and sample it at the beginning of each patrolling stage. The assumption underlying this point of view is that the poacher is a strategically sophisticate actor: The poacher will spy on the ranger and will learn whichever probability distribution she has previously used. Afterwards, the poacher will choose a route along the trails which is most likely to avoid the ranger. Assuming this informational advantage for the poacher, the ranger can choose a distribution that forces the poacher to adopt the least convenient optimal response. First, we will explore the general preliminaries of Stackelberg games that will serve to our main purpose

2.1 General preliminaries

A Stackelberg (after Von Stackelberg 1934) game is a game played sequentially between two players: the *leader* and the *follower*. The leader commits to a strategy, the follower observes the strategy adopted by the leader and after that commits to his own strategy. A Stackelberg security game (SSG) is a special case. In these the leader is called the *defender* and the follower is called the *attacker*.

In an SSG the defender must perpetually defend a finite set T of targets using a limited number of resources. A *pure strategy* for the defender consists of deploying a set R of resources by selecting a subset of T which becomes guarded. A *mixed strategy* for the defender consists of a probability distribution over the set of pure defender strategies.

A *pure strategy* for the attacker consists of attacking a target $t \in T$. Each target has a specified set of payoff values for both the defender and the attacker. This payoff depends *exclusively* on whether or not t was being guarded and NOT on whether other non-attacked targets where guarded or not.

So to specify payoffs completely it suffices to have two matrices (with payoffs for defender and attacker resp.) with rows corresponding to $t \in T$ and only two $GUARDED/UNGUARDED$ columns. We represent these as follows. For $t \in T$

1. Defender payoff for each target t, either uncovered $U_d^u(t)$ or covered $U_d^g(t)$
2. Attacker payoff for each target t, either uncovered $U_a^u(t)$ or covered $U_a^g(t)$

For simplicity we will begin by assuming that the defender can guard only one target

$t \in T$ (i.e. that $|R| = 1$). The game is now specified by the following utility functions of defender and attacker with $(a, d) \in T \times T$:

$$U_d(a, d) = \sum_{t \in T} (U_d^g(t) 1_{t=a, t=d} + U_d^u(t) 1_{t=a, t \neq d})$$

and

$$U_a(a, d) = \sum_{t \in T} (U_d^g(t) 1_{t=a, t=d} + U_d^u(t) 1_{t=a, t \neq d})$$

where 1_X denotes the characteristic function of a set X.

These utility functions already incorporate the leader-follower dynamic in the sense that $U_a(a, d)$ (resp $U_d(a, d)$) should be interpreted as the payoff received by the attacker (resp. the defender) when the defender defends target d and *afterwards* the attacker hits target a.

If D and A are random variables ranging over T representing mixed strategies for attack and defense then we write $U_j(A, D) := \mathbb{E}[U_j(A, D)]$ for $j \in \{a, d\}$ where A and D are assumed to be independent random variables. This independence does not contradict the fact that A occurs after D because it speaks about the sources of randomness that underlie the random assortment of pure strategies.

We write $U_j(a, D)$ to mean $U_j(A, D)$ where A is the Dirac delta centered at $a \in T$.

If α_t and β_t denote the probability densities of A and D we have

$$U_j(A, D) = \sum_{t \in T} \alpha_t U_j(t, D) = \sum_{t \in T} \alpha_t \left(U_j^g(t) \beta_t + U_j^u(t)(1 - \beta) \right)$$

And in particular $U_j(A, D)$ is a bilinear function of α and β.

2.2 Stackelberger equilibria

The leader-follower dynamic implies that in the first turn the defender chooses a strategy D with distribution β and in the second turn the attacker picks a target using a distribution α which may depend on β. Since the attacker wishes to maximize his utility determining the attacker's distribution α amounts to solving the following problem

$$\max_{\alpha_t} \sum_{t \in T} \alpha_t U_a(t, D) \tag{1}$$

$$s.t. \sum \alpha_t = 1, \alpha_t \geq 0 \tag{2}$$

Since D (with distribution β_t) is given, the quantities

$$U_a(t, D) = U_a^g(t)\beta_t + U_a^u(t)(1 - \beta_t)$$

are known constants and therefore the Problem (2.2) that the attacker has to solve is a linear optimization problem over the probability simplex. Since the objective function is linear its maximum is achieved at an extreme point of the simplex and therefore we can always assume that there exists an optimal response of the attacker which is a pure strategy. We call such a strategy a response to the defender strategy D and denote it by $t(D)$. More precisely the response function $t(D)$ is defined as an argmax of problem (2.2) given D.

Now, how should the defender choose the strategy D? In order to maximize the resulting expected payoff it should therefore find D which maximizes the final defender utility taking into account the way in which the (in the attacker's view optimal, given D) attacker will respond. The defender wishes to solve

$$\max_D U_d(t(D), D).$$

Definition 2.1. A pair (t^*, D^*) is an equilibrium if:

1. $U_d(t^*, D^*) \geq U_d(t(D), D)$ for every mixed strategy D.
2. $t = t^* \in T$ should be a maximizer of $U_a(t, D^*)$.

The dual of problem (2.2) is given by

$$\min_{\lambda \in \mathbb{R}} \lambda \text{ s.t.:} \tag{3}$$

$$\forall t \in T \, (\lambda \geq U_a(t, D)) \tag{4}$$

By weak duality we know that for every pair (α, λ) feasible in the primal and dual the inequality

$$\sum \alpha_t U_a(t, D) \leq \lambda$$

holds. Furthermore we know that equality occurs if and only if (α, λ) is a pair of optimal solutions of primal and dual respectively.

This duality relationship can be used to give a mixed-integer quadratic optimization formulation to the problem of finding equilibria. The key point is that the constraints ensure that variables (α_t, β_t) represent an arbitrary probability distribution β for the defender and

an *optimal* response α for the attacker when the defender uses the mixed strategy with density β_t.

Theorem 2.2. *Let M be a fixed sufficiently large real number. If $(\alpha^*, \beta^*, \lambda^*)$ is an optimal point for the optimization problem below then (α^*, β^*) is a Stackelberg equilibrium for the leader-follower game.*

$$\max_{\lambda \in \mathbb{R}, \alpha \in \{0,1\}^T, \beta \in \mathbb{R}^T} U_d(\alpha, \beta) \text{ s.t.:} \tag{5}$$

$$\sum \alpha_t = 1, \beta_t, \alpha_t \geq 0, \sum \beta_t = 1 \tag{6}$$

$$\forall t \in T \, (\lambda \geq U_a(t, \beta)) \tag{7}$$

$$\forall t \in T \, (M(1 - \alpha_t) + U_a(t, \beta) \geq \lambda) \tag{8}$$

Proof. Note that α_t can only assume the discrete values $\{0, 1\}$ while β_t assumes real non-negative values with $\sum \beta_t = 1$. Note also that the objective function is quadratic (in fact bi-linear) in (β, α) and that all constraints are linear equations on variables which are either real or integer. Equation (6) says that α and β are probability distributions. Equation (7) guarantees that λ is an upper bound for the optimization problem of the follower. The most interesting constraint is equation (8) which we claim *forces* α to be an optimal response for the attacker. This is because if (α, β, λ) are a feasible point then there exists a unique $t^* \in T$ such that $\alpha_{t^*} = 1$. It follows that $U_d(\alpha, \beta) = U_d(t^*, \beta)$ and by inequality (8) that $U_a(t^*, \beta) \geq \lambda$ so λ is an optimal value for the attacker problem with defense distribution β and the corresponding α are an optimal response for the attacker as claimed. We conclude that an optimal point for the optimization problem produces a Stackelberg equilibrium (α^*, β^*) as claimed and furthermore λ equals the payoff to the attacker for the optimal response to the defense α^*. □

Remark 2.3. By the compactness and non-emptiness of the feasible region and the continuity of the objective function in the previous formulation we conclude that every Stackelberg game has at least one equilibrium. Furthermore the equilibrium above has the strange additional property that if there are several optimal responses for the attacker with the same payoff then he chooses the one that maximizes the defender's payoff.

2.3 Ranger's problem as a Stackelberg Security Game

We want to model the Ranger's problem using the leader and follower dynamic of a Stackelberg Security Game. In this particular framework, the set of targets will be the set of

feasible routes for the ranger described by C_G, the matrix of cycles of the graph of trails G. Therefore, a mixed strategy will be a probability distribution over the columns of C_G.

Let $t \in \{0,1\}^n$ be the route of the poacher and suppose that C^j is the column of C_G chosen by the ranger. The reward for the ranger given the agent's choices is the number of trails in which they meet minus the number of trails that the ranger left unguarded but the poacher doesn't

$$U_r(t, C^j) = \sum_{i=1}^{n} t_i C_i^j - (1 - C_i^j)t_i$$

Similarly, the reward of the poacher is

$$U_p(t, C^j) = \sum_{i=1}^{n} (1 - C_i^j)t_i - t_i C_i^j = -U_r(t, C^j)$$

Here we changed the subscripts a and d (after atacker and defender) for p and r (for poacher and ranger).

The expected reward of the ranger after choosing a route t provided the poacher walks along the cycle C^j with probability ρ_j is then

$$U_p(t, \rho) = \sum_{j=1}^{M} \rho_j U_p(t, C^j)$$

Since the poacher is spying over the ranger she will pick the route at random (fix a probability distribution), acknowledging the ranger's decision ρ . Namely, given the ranger distribution ρ, the poacher will respond with the distribution $t(\rho)$ obtained by solving

$$\max_{\pi \in \mathbb{R}^{\#\ R_p}} \sum_{t \in R_p} \pi_t U_p(t, \rho) \tag{9}$$

$$\text{s.t.:} \tag{10}$$

$$\pi_t \geq 0 \tag{11}$$

$$\sum \pi_t = 1 \tag{12}$$

where $R_p \subseteq \{0,1\}^n$ is some set of feasible routes for the poacher. The problem 9 is a linear program over a probability simplex and thus it has solution in an extreme point of the form $(0, \cdots, 1, \cdots, 0)^T$ meaning that $t(\rho)$ is the route in R_p corresponding to the only $\alpha_t > 0$.

Taking this information into account the ranger chooses ρ so as to maximize $U_r(t(\rho), \rho)$.

$$\max_{t,x\in\mathbb{R}^n} U_r(t(\rho),\rho)$$
$$\text{s.t.:}$$
$$\rho \geq 0$$
$$\sum \rho_j = 1$$

An optimal solution (ρ^*, t^*) to 2.3 is a Stackelberg equilibrium. The next section will focus on describing this problem using concise algebraic constraints using the big M formulation of the problem (2.2 , [2]) and ideas from [5].

2.4 The circuit Polytope

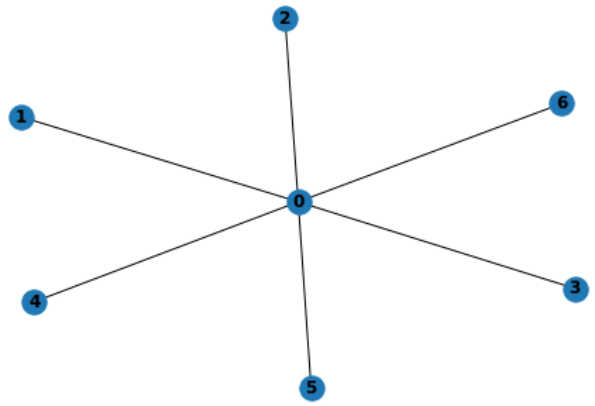
In 1997 Petra Bauer [5] gave, for the complete graph G, the inequalities characterizing the set of integer points $\mathcal{C}_G$ of its circuit polytope P_G. Note that the integer points of P_G when G is the graph of trails are precisely the columns of C_G. This suffices for describing $\mathcal{C}_G$ for an arbitrary graph since the only thing left to do is to include the restrictions related to the topology of the graph G embedded in the complete graph of $|V(G)|$ nodes. For doing so, we will define the incidence matrix of an arbitrary graph as usual but taking into account all the possible edges that G might have. More precisely, for a directed graph $G = (E, V)$ its boundary matrix $\partial_G \in \mathbb{R}^{|V|\times|E|}$ is usually defined by

$$(\partial_G)_{ij} \begin{cases} 1 & e_j[0] = v_i \\ -1 & e_j[1] = v_i \\ 0 & \text{elsewhere} \end{cases}$$

where e_j denotes the $j-$th edge and v_i denotes the $i-$th node. The incidence matrix δ_G of G is the component-wise absolute value of ∂_G. Note that δ_G is independent of the orientation of G so we will use it for both oriented and graph without orientation.

As an example, consider the complete graph K_n. In this case, δ_{K_n} has n rows and $\binom{n-1}{2}$ columns since $E = \{\{r, s\} : 1 \leq r, s \leq n\}$. We will enumerate the edges of the complete graph using the biyection $b(r, s)$ given by

$$\{r, s\} \mapsto \sum_{k=1}^{r-1} n - l + s - r = n(r-1) - \frac{(r-1)r}{2} + s - r$$



$$\partial_G = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

Boundary matrix of the star graph of 6 edges

for $1 \leq r < s \leq n$. Under this convention, the first edge is $\{1, 2\}$ and the last one is $\{n-1, n\}$. This enumeration corresponds to enumerate the upper triangular part of a $n \times n$ matrix from left to right, top to bottom.

In this document, instead of defining δ_G as a matrix with $|V|$ rows and $|E|$ columns, it will be a matrix with $|V|$ rows and $\binom{|V|-1}{2}$ columns. If no edge in G connects the nodes s and r $(r < s)$, then the $b(r, s)$ column is zero everywhere. Otherwise, the $b(r, s)$ column is zero everywhere except for the entries s and r. This allows us to extend Bauer's inequalities to an arbitrary graph. Intuitively, to check if $t \in \mathbb{R}^{\binom{|V|-1}{2}}$ is the incidence vector of a circuit in G ones has to check the following:

1. **Non-negativity and integer constraints:** C has non-negative and integer entries.
2. **Parity and degree constraints:** Every edge traversed by the path C has exactly two distinct adjacent edges traversed by C. Or equivalently, C traverses every node at most 2 times (degree) and if it traverses the node, it traverses twice (parity).
3. **Connectedness constraints:** C does not corresponds to a disjoint union of circuits.
4. **Length constraints:** C traverses at least 3 distinct edges.
5. **Topology constraints:** C is zero in every entry that corresponds to edges non present in G.

Petra Bauer proved that these conditions (except for the last one) are not only necessary but also sufficient for x being equal to the incidence vector of a circuit over the complete graph. The last constraint is the only one remaining to extend her results to arbitrary graphs. Even though intuitive, the connectedness constraint is demanding to write and it requires to introduce new definitions and notation before explicitly state Bauer's theorem.

Definition 2.4.

- A cut of a graph is a two element partition of the set of nodes i.e. $S \subseteq V$, $V \setminus S$.
- Given two disjoints sets $S, T \subseteq V$, we denote by $(S : T)$ the incidence vector of the set of edges with one node in S and the other one in T.
- Given a set of nodes $S \subseteq V$ its incidence vector $\delta_G(S)$ is defined as the incidence vector of the set $(S : V \setminus S)$. The vector $\delta_G(S)$ is called a cut as well.

The previous notation does not interfere with the notation of the incidence matrix δ_G. If $\delta_G(i)$ is the i-th row of δ_G and v is the $i-$th node of G then $\delta_G(i) = \delta_G(\{v\})$. From now on we will denote both by $\delta_G(v)$.

Theorem 2.5 (Bauer)**.** *Let $G = (E, V)$ be a graph and δ_G its incidence matrix. Then $\mathcal{C}_G$ is the set of solutions to the following system of linear inequalities*

1. ***Nonnegativity and integer constraints:*** *$t \geq 0$ and $t \in \mathbb{Z}$.*
2. ***Degree constraints:*** *$\delta_G t \leq 2\mathbb{1}$.*
3. ***Parity constraint:*** *$\langle x, \delta_G(v) \rangle \geq 2\langle x, e_i \rangle$ for every $v \in V$ and every i for which $\delta_G(v)_i \neq 0$ (here e_i is the $i-$th canonical vector)*
4. ***Connectedness constraints:*** *$t_e + \langle t, (e[0] : S) \rangle - \langle t, (S : T) \rangle \leq 2$ for all $e \in E$ and for all S, T partitioning $V \setminus \{e[0], e[1]\}$.*
5. ***Length constraints:*** *$\langle t, \mathbb{1} \rangle \geq 3$*
6. ***Topology constraints:*** *$t \leq \mathbb{1}^T \delta_G$.*

Proof. t equals the incidence vector of some circuit on the complete graph if and only if it satisfies the first 5 inequalities (Theorem 3.8 [5]). A circuit in the complete graph is a circuit over G if and only if $t_e = 0$ for every $e = (i, j) \notin E$. Since all columns of δ_G sum up either to 2 or 0 then t is a circuit on the complete graph satisfying the nonnegativity constraint and the topology constraint if and only if $t_e \neq 0$ only in the corresponding columns having at least one non-zero entry. □

Theorem 2.2 proves that in general frameworks an Stackelberg equilibria can be calculated by means of solving a mixed integer quadratic program with $2 * T + 1$ variables. Our first result is the following theorem which proves that in this particular instance where the targets are cycles over the graphs of trails, the Stackelberg Equilibria can be calculated by means of a integer quadratic program with only $2 * |V(G)| + 1 = 2n + 1$ variables, where G is the graph of trails.

Theorem 2.6 (M. Velasco, N. Betancourt)**.** *Consider the problem*

$$\begin{aligned} &\max_{\lambda\in\mathbb{R},\ x,t\in\mathbb{R}^n} \sum_{i=1}^{n} t_i x_i - (1-x_i)t_i \\ &\qquad\qquad s.t.: \\ &\qquad x \in P_G,\ t \in \mathcal{C}_G \\ &\qquad \forall t\prime \in \mathcal{C}_G \left(\lambda \geq U_p(t\prime, x)\right)) \\ &\forall t\prime \in \mathcal{C}_G \left(M \operatorname{Tr}\left((\mathbb{1} - t\prime)t^T\right) + U_p(t\prime, x) \geq \lambda\right) \end{aligned}$$

For a large enough $M \in \mathbb{N}$ this problem has a solution. In that case, for every solution (λ^*, x^*, t^*) the pair (x^*, t^*) is a Stackelberg Equilibrium.

Proof. The expected ranger's utility of fixing a distribution ρ provided the poacher follows a path t is

$$\begin{aligned} U_r(t,\rho) &:= \sum_{j=1}^{M} \rho_j U_r(t, C^j) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{M} \rho_j \left(t_i C_i^j - (1 - C_i^j)t_i\right) \\ &= \sum_{i=1}^{n} t_i \left(\sum_{j=1}^{M} \rho_j C_i^j\right) - \left(1 - \left(\sum_{j=1}^{M} \rho_j C_i^j\right)\right) t_i \\ &= \sum_{i=1}^{n} t_i (x_\rho)_i - (1 - (x_\rho)_i)t_i \end{aligned}$$

where x_ρ is the average cycle of the distribution ρ. Meaning that the expected utility $U_a(t,\rho)$ is the utility of t and the average cycle x_ρ . Note that x_ρ is a point in the convex closure of the columns of C_G i.e. $x_\rho \in P_G$

Assume that the number $\operatorname{Tr}\left((\mathbb{1} - t')t^T\right)$ is zero if and only if $t = t\prime$ and a positive integer in any other case. Then, if (λ, x, t) is a feasible point we have

$$U_a(t,x) = M \operatorname{Tr}\left((\mathbb{1} - t)t^T\right) + U_a(t,x) \geq \lambda \geq U_a(t,x)$$

i.e. $U_a(t,x)$ is an upper bound of $\{U_a(t\prime, x)\}_{t\prime\in\mathcal{C}_G^a}$ and t is a maximizer. If additionally the

feasible point is a solution, (t, x) is a Stackelberg equilibrium.

This shows that it suffices to prove the first statement i.e. $\mathrm{Tr}\left((\mathbb{1} - t')t^T\right) = 0 \Leftrightarrow t = t\prime$.

If $t = t\prime$ then for every j we have $1 - t'_j \neq 0 \Leftrightarrow t_j = 0$ and therefore $\mathrm{Tr}\left((\mathbb{1} - t)t^T\right) = \sum_j (1 - t'_j)t_j = 0$.

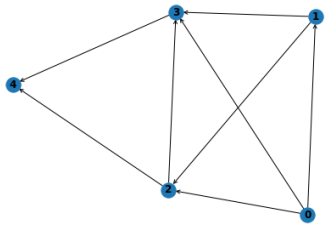
Now assume $\mathrm{Tr}\left((\mathbb{1} - t')t^T\right) = \sum_j (1 - t'_j)t_j = 0$. Since $0 \leq t'_j, t_j \leq 1$ for every j, this can happen if and only if $(1 - t'_j)t_j = 0$ for every j. From this it follows that $t_j = 1 \Rightarrow t'_j = 1$ and $t'_j = 0 \Rightarrow t_j = 0$ i.e. the circuit of t' contains the circuit of t. In order to arrive to a contradiction assume that there is a j for which $t'_j = 1$ and $t_j = 0$.

If v is a node adjacent with j then there is a $j_1 \in t' \setminus \{j\}$ adjacent with v due to the parity constraint in the node v. If $j_1 \in t$ then there is a $l_1 \in t \setminus \{j_1\}$ adjacent with v which is different from j. By hypothesis we have $l_1 \in t'$ but this would lead to a contradiction with the degree constraints in the node v. We conclude that $j_1 \in t' \setminus t$. Let $w \neq v$ be a node adjacent with j_1. We can argue as before to find a $j_2 \notin \{j_1, j\}$ adjacent with w such that $j \in t' \setminus t$. This process will eventually stabilize i.e. there is a maximum $m \in \mathbb{N}$ $(m \geq 3)$ such that we can build a chain $j_0 := j, j_1, \cdots, j_m \in t' \setminus t$ of distinct vertices such that j_l and j_{l+1} are adjacent. Let v_m be the node that is common to j_{m-1} and j_m. If $w_m \neq v_m$ is the remaining node to which j_m is adjacent then there is a vertex $j_{m+1} \in t' \setminus t$ adjacent to w_m. By the maximality of m, $j_{m+1} \in \{j_0, j_1, \cdots, j_m\}$. In fact $j_{m+1} \neq j_m$ due to the parity constraint on w_m. Additionally, $j_{m+1} \neq j_l$ for $l > 0$, otherwise j_{m+1} would be adjacent to two distinct vertices different from j_m and it would imply that there is a node violating its degree constraints. This proves that $j_{m+1} = j_0$ i.e. $j, j_1, \cdots, j_m$ is a circuit contained in t' which is disjoint to t. This is a contradiction with the fact that t' satisfies the Connectedness constraints.

□

2.5 Computational examples

The house graph.



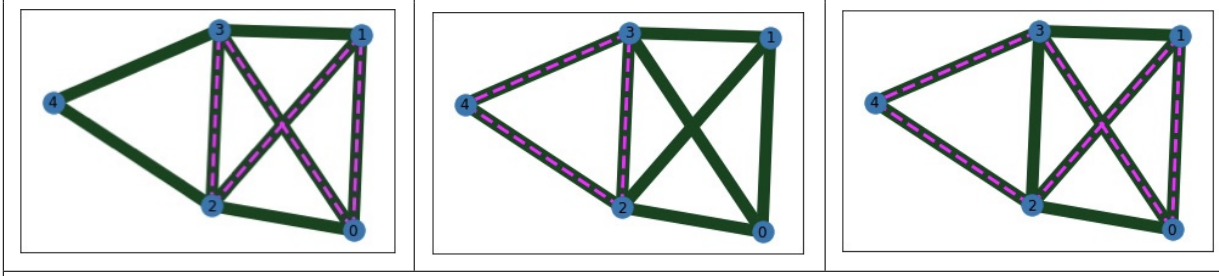
To conclude this section we provide a computational example. Here a ranger is opposed to a Stackelrberg-like tactical poacher trying to perform illegal activities within the area of a

synthetically reserve with a planar graph of trail of $n = 5$ nodes known as the house graph. Since the graph in the example is small and there are very few circuit like paths, we will ignore the constraint of the circuits crossing an specific node.

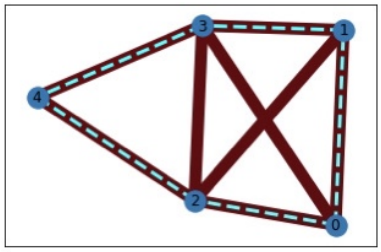
We computed the Stackelberg Equilibria by solving the corresponding instance of 2.6 and obtained an optimal distribution for the ranger having a support of 3 cycles. It is noteworthy that this choice of ρ forces the poacher to have an inconvenient optimal response π that meets the ranger's cycles always

Ranger's equilibria distribution ρ

$\rho_j 0.6$ $\rho_j = 0.2$ $\rho_j = 0.2$



Poacher's optimal response π



To provide an additional interesting computational example note that Problem 2.6 can be modified so as to assign each trail i (edge on the graph) four weights:

- A weight indicating how inconvenient it is for the ranger to left trail i unguarded (X_{i1}).
- A weight indicating how convenient it is for the ranger to guard trail i (X_{i2}).
- A weight indicating how inconvenient it is for the poacher to be caught on trail i Y_{i1} .
- A weight indicating how convenient it is for the poacher to perform its illegal activities along trail i (Y_{i2}).

This amounts only to set some coefficient in the functions U_p and U_r.

$$U_r(t, C^j) = \sum_{i=1}^{n} X_{i1} t_i C_i^j - X_{i2}(1 - C_i^j) t_i$$

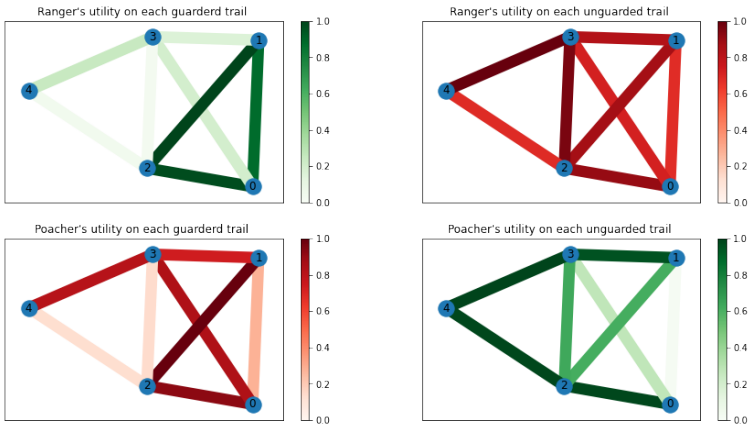
And

$$U_p(t, C^j) = \sum_{i=1}^{n} Y_{i2}(1 - C_i^j)t_i - Y_{i1}t_i C_i^j$$

For instance, our original reward functions correspond to the case $X_{i1} = X_{i2} = 1$ and $Y_{i1} = Y_{i2} = 1$. Note that re relation $U_p(t, C^j) = -U_r(t, C^j)$ will not longer hold necessarily.

In the following example we assigned those weights at random ($\sim \mathcal{N}(0, 0.8)$). Here, a visualization of the resulting weight assignment

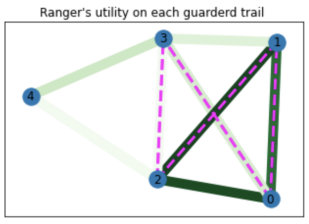
The weighted house graph for each agent.



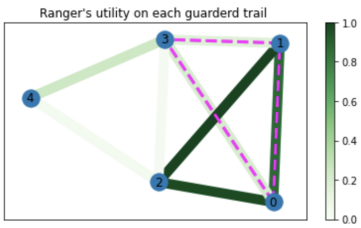
In this particular case the distribution ρ of the ranger has a 5 cycles support depicted below

Ranger's equilibria distribution ρ in the weighted case

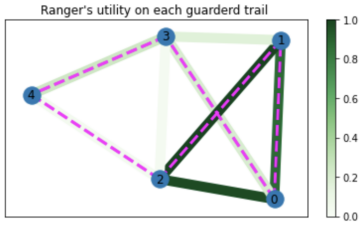
$\rho_j = 0.43$



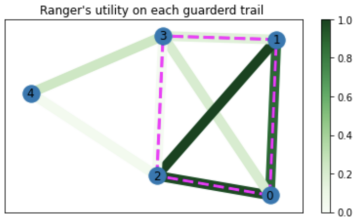
$\rho_j = 0.25$



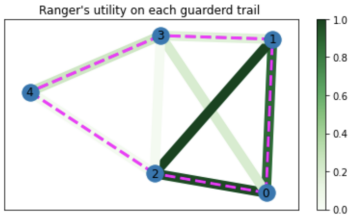
$\rho_j = 0.23$



$\rho_j = 0.05$



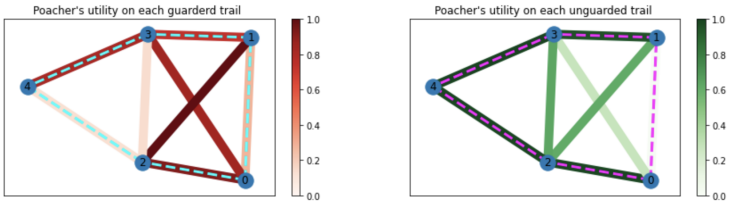
$\rho_j = 0.04$



After learning the distribution ρ, the poacher reacts with the route that avoids the ranger

the most taking into account the weights

Poacher's optimal response π.



What is interesting about this weighted scenario is that the poacher is completely cornered. Note that the resulting optimal poacher response is equal to the cycle in the support of ρ with least patrolling frequency.

3 Combinatorial Multi-armed Bandits for combinatorial stochastic optimization problems

The game-theoretical approach described in the previous section presents two serious difficulties: First the assumptions about the poacher might not be true causing the suggested routes to be overly conservative and perform badly. And second, optimizing over the space of probability distributions on the cycles of a given graph turns out to be computationally demanding and infeasible for cases of practical interest.

The last section of this document will introduce a method that is computationally more practical and also makes weaker, and in our view more realistic, assumptions on the behavior of the poacher. The idea is to apply tools from decision making under uncertainty as done in [3]. In this chapter we introduce the concept of Multi-armed Bandits and its combinatorial version which are fundamental for understanding the solution to the ranger's problem using online-learning.

3.1 Multi-armed bandits

Sequential decision making under uncertainty encompasses a wide variety of problems. In its most plain scenario, an agent is presented with a set of available controls or actions (also known as experts). Each time that a given control is chosen, the agent perceives a random outcome that can be interpreted as a reward or penalty depending on the context. The goal of the agent is to fix a decision-making policy so as to maximize (or minimize in the penalty point of view) his cumulative reward after a budgeted number of trials. This very simple instance is known as multi-armed bandit and despite its generality is very popular for it is useful for modeling problems in finance ([6]), medical trials ([7]), route programming ([8]) and many more ([9],[10],[11]).

More concisely a K-armed bandit is a process determined by a fixed amount of allowed trials or times $t = 0, 1, \ldots, T$ and a set of random variables $X_1, \cdots, X_K$, called arms, with finite means $\mu_i := \mathbb{E}[X_i]$ $(1 \leq i \leq K)$. The distribution of the arms are unknown and at each time $t = 0, 1, \ldots, T$ the agent hast to sample on of them perceiving a reward $x_t^{i_t}$.

If $i^* := \text{argmax}\{\mu_1, \cdots, \mu_K\}$ and $\mu^* := \max\{\mu_1, \cdots, \mu_K\}$ then the decision rule that maximizes the expected cumulative reward of the agent after T stages is always choosing the i^*-th arm. This constant decision results in a expected cumulative reward of $T\mu^*$. Nonetheless, always sampling the i^*th arm is unfeasible, for the distribution of the arms are unknown. In view of this, the best course of action is trying to estimate the expected value

of all of the arms, which faces one primary difficulty: the natural thing to do is to use the Law of Large number to obtain an approximation of each μ_i converging almost surely to the actual expected value. The limited budget of trials available (T) might be less than enough to approximate each μ_i properly, for the convergence might be slow. From this dilemma came the need of a decision rule that attempts to estimate the μ_i's, spending most of the resources in the arms that seem to have larger average return when sampled i.e. an algorithm balancing the sampling of poorly estimated arms (exploration) and exploitation, which is choosing the arm with the highest estimated mean so far.

This dilemma motivates the definition of *regret*, which is a measure that quantifies how well a fixed method is doing. The regret at stage t is defined by

$$t\mu^* - \mathbb{E}\left[\sum_{s=0}^{t} x_s^{i_s}\right]$$

where $x_s^{i_s}$ is the observed reward at stage s resulting from sampling arm i_s suggested by a fixed policy. Thus, the goal of our agent is to fix a decision rule that minimizes the regret which is equivalent to maximize the expected cumulative reward.

Asymptotic bounds on the regret can be given making few assumptions on the arms as soon as a policy is fixed. There are other criteria for measuring policies performance including the probability of selecting the optimal arm, the expected number of times in which that arm is chosen among others. The following section will be devoted to describe a method for balancing exploration and exploitation. We will follow [12]. In the article they use a well known result in statistics , namely, Chernoff inequality.

Theorem 3.1 (Chernoff inequality)**.** *Let $X_1, \cdots, X_k$ be random variables supported in a common subset of $[0,1]$. If $\mu := \mathbb{E}\{X_k | X_1, \cdots, X_{k-1}\}$ and $\overline{X}_k := \frac{1}{k}\sum^k X_i$ then for every $a > 0$*

$$\mathbb{P}\{\mu - \overline{X}_k \geq A\} \leq \exp(-2kA^2) \quad \mathbb{P}\{\overline{X}_k - \mu \geq A\} \leq \exp(-2kA^2)$$

A proof of theorem 3.1 can be found in [13] Section 2.2 .

3.1.1 Upper confidence bound policy

Now we will introduce Upper confidence bound, a central algorithm throughout this document. Upper confidence bound takes in to account the poorly estimated arms so far and the ones that seem promising candidates for optimal arm. This is achieved by sequentially selecting the arm maximizing a quantity that increases with both, the value of the current estimate of the mean and the reciprocal of the number of times an arm has been selected.

More precisely, given a parameter $c > 0$, Upper confidence bound policy chooses an arm i_t at time t according to the rule

$$i_t = \underset{i=1,\cdots,K}{\arg\max} \left\{ \overline{X}_{i,T_i(t-1)} + c\sqrt{\frac{\ln k}{T_i(t-1)}} \right\}$$

where $T_i(t)$ is the number of times arm i has been played up to stage t and $\overline{X}_{i,n} := \frac{1}{n}\sum_{t=1}^{n} x_t$ is the sample average of the first n observations of the ith arm. The parameter $c > 0$ let us establish priority between exploration and exploitation.

Bellow we state and proof a bound of $T_i(t)$ for $i \neq i^*$, which quantifies the rate at which the number of mistakes might increase

Theorem 3.2. *If the upper confidence bound policy is implemented then for every sub-optimal arm i, the expected value of the number of times that arm will be chosen up to stage t is at most*

$$4c^2 \frac{\ln(t)}{(\mu_i - \mu^*)^2} + 1 + 2\varphi(2c^2 - 2)$$

provided that the arms have common support in $[0, 1]$. Here, $\varphi(z)$ is the Riemann Zeta Function.

Proof. Let I_t be the random variable representing the arm index played at time t. Also following [12] notation, we will use $*$ in the previous symbols to denote the variables corresponding to the optimal arm: $T^*(k), \overline{X}^*$.

Note that at the end of stage K every arm has been chosen exactly once. Now fix an arm i other than the optimal. For any integer $l > 1$ and any time index $t > K$ if $T_i(t) \geq l$ let t_l be the time index at which $T_i(t_l) = l$. Note that $t_l \leq t$ provided the assumption $T_i(t) \geq l$. After t_l up to t there were exactly $T_i(t) - l$ stages at which $I_s = i$, or, symbolically

$$\sum_{s=K+1}^{t} \mathbb{1}\{I_s = i, T_i(s-1) \geq l\} = T_i(t) - l$$

equivalently

$$T_i(t) = l + \sum_{s=K+1}^{t} \mathbb{1}\{I_s = i, T_i(s-1) \geq l\}.$$

If , otherwise, $T_i(t) < l$ trivially we have that

$$T_i(t) < l + \sum_{s=K+1}^{t} \mathbb{1}\{I_s = i, T_i(s-1) \geq l$$

Hence, in every possible case the inequality

$$T_i(t) \leq l + \sum_{s=K+1}^{t} \mathbb{1}\{I_s = i, T_i(s-1) \geq l\}$$

holds

Since $I_s = i$ implies $\overline{X}^*_{T^*(s-1)} + c\sqrt{\frac{\ln(s-1)}{T^*(s-1)}} \leq \overline{X}_{i,T_i(s-1)} + c\sqrt{\frac{\ln(s-1)}{T_i(s-1)}}$ then

$$\begin{aligned} T_i(t) \leq & l + \sum_{s=K+1}^{t} \mathbb{1}\left\{I_s = i, T(s-1) \geq l\right\} \\ \leq & l + \sum_{s=K+1}^{t} \mathbb{1}\left\{\overline{X}^*_{T^*(s-1)} + c\sqrt{\frac{\ln(s-1)}{T^*(s-1)}} \leq \overline{X}_{i,T_i(s-1)} + c\sqrt{\frac{\ln(s-1)}{T_i(s-1)}}, T(s-1) \geq l\right\} \end{aligned}$$

Since $0 < T^*(s-1) < s$ we have that $\overline{X}^*_{T^*(s-1)} + c\sqrt{\frac{\ln(s-1)}{T^*(s-1)}} \in \left\{\overline{X}^*_r + c\sqrt{\frac{\ln(s-1)}{r}} \ : \ 0 < r < s\right\}$ hence

$$\begin{aligned} T_i(t) \leq & l + \sum_{s=K+1}^{t} \mathbb{1}\left\{I_s = i, T(s-1) \geq l\right\} \\ \leq & l + \sum_{s=K+1}^{t} \mathbb{1}\left\{\overline{X}^*_{T^*(s-1)} + c\sqrt{\frac{\ln(s-1)}{T^*(s-1)}} \leq \overline{X}_{i,T_i(s-1)} + c\sqrt{\frac{\ln(s-1)}{T_i(s-1)}}, T(s-1) \geq l\right\} \\ \leq & l + \sum_{s=K+1}^{t} \mathbb{1}\left\{ \min_{0<r<s}\left\{\overline{X}^*_r + c\sqrt{\frac{\ln(s-1)}{r}}\right\} \leq \overline{X}_{i,T_i(s-1)} + c\sqrt{\frac{\ln(s-1)}{T_i(s-1)}} \ , T(s-1) \geq l\right\} \\ \leq & l + \sum_{s=K+1}^{t} \mathbb{1}\left\{ \min_{0<r<s}\left\{\overline{X}^*_r + c\sqrt{\frac{\ln(s-1)}{r}}\right\} \leq \max_{l\leq v<s}\left\{\overline{X}_{i,v} + c\sqrt{\frac{\ln(s-1)}{v}}\right\} \ \right\} \end{aligned}$$

If for every $0 < r < s$ and every $l \leq v < s$ we have $\overline{X}^*_r + c\sqrt{\frac{\ln(s-1)}{r}} > \overline{X}_{i,v} + c\sqrt{\frac{\ln(s-1)}{v}}$ then

$$\min_{0<r<s}\left\{\overline{X}^*_r + c\sqrt{\frac{\ln(s-1)}{r}}\right\} > \max_{l\leq v<s}\left\{\overline{X}_{i,v} + c\sqrt{\frac{\ln(s-1)}{v}}\right\}$$

Thus, the event

$$\min_{0<r<s}\left\{\overline{X}^*_r + c\sqrt{\frac{\ln(s-1)}{r}}\right\} \leq \max_{l\leq v<s}\left\{\overline{X}_{i,v} + c\sqrt{\frac{\ln(s-1)}{v}}\right\}$$

implies that there exist a pair of time indexes $0 < r < s$ and $l \leq v < s$ such that $\overline{X}_r^* + c\sqrt{\frac{\ln(s-1)}{r}} \leq \overline{X}_{i,v} + c\sqrt{\frac{\ln(s-1)}{v}}$

$$\begin{aligned} T_i(t) \leq & l + \sum_{s=K+1}^{t} \sum_{r=1}^{s-1} \sum_{v=l}^{s-1} \mathbb{1} \left\{ \overline{X}_r^* + c\sqrt{\frac{\ln(s-1)}{r}} \leq \overline{X}_{i,v} + c\sqrt{\frac{\ln(s-1)}{v}} \right\} \\ \leq & l + \sum_{s=1}^{\infty} \sum_{r=1}^{s-1} \sum_{v=l}^{s-1} \mathbb{1} \left\{ \overline{X}_r^* + c\sqrt{\frac{\ln s}{r}} \leq \overline{X}_{i,v} + c\sqrt{\frac{\ln s}{v}} \right\} \end{aligned}$$

If all of the following hold

$$\begin{aligned} \overline{X}_r^* &> \mu^* - c\sqrt{\frac{\ln s}{r}} \\ \overline{X}_{i,v} &< \mu_i + c\sqrt{\frac{\ln s}{v}} \\ \mu^* &\geq \mu_i + 2c\sqrt{\frac{lns}{v}} \end{aligned}$$

then

$$\overline{X}_r^* + c\sqrt{\frac{\ln s}{r}} > \mu^* \geq \mu_i + 2c\sqrt{\frac{lns}{v}} > \overline{X}_{i,v} + c\sqrt{\frac{\ln s}{v}}$$

It follows that for $\overline{X}_r^* + c\sqrt{\frac{\ln s}{r}} \leq \overline{X}_{i,v} + c\sqrt{\frac{\ln s}{v}}$ to be true, on of te following must hold

$$\begin{aligned} \overline{X}_r^* &\leq \mu^* - c\sqrt{\frac{\ln s}{r}} \\ \overline{X}_{i,v} &\geq \mu_i + c\sqrt{\frac{\ln s}{v}} \\ \mu^* &< \mu_i + 2c\sqrt{\frac{lns}{v}} \end{aligned}$$

Using this in our bound for $T_i(t)$ we obtain

$$\begin{aligned}
T_i(t) \leq & l + \sum_{s=1}^{\infty}\sum_{r=1}^{s-1}\sum_{v=l}^{s-1} \mathbb{1}\left\{ \overline{X}_r^* + c\sqrt{\frac{\ln s}{r}} \leq \overline{X}_{i,v} + c\sqrt{\frac{\ln s}{v}} \right\} \\
\leq & l + \sum_{s=1}^{\infty}\sum_{r=1}^{s-1}\sum_{v=l}^{s-1} \mathbb{1}\left\{ \mu^* - \overline{X}_r^* \geq c\sqrt{\frac{\ln s}{r}} \right\} \\
& + \mathbb{1}\left\{ \overline{X}_{i,v} - \mu_i \geq c\sqrt{\frac{\ln s}{v}} \right\} \\
& + \mathbb{1}\left\{ \mu^* < \mu_i + 2c\sqrt{\frac{lns}{v}} \right\}
\end{aligned}$$

Applying the expected value operator to the inequality we obtain

$$\begin{aligned}
\mathbb{E}[T_i(t)] \leq & l + \sum_{s=1}^{\infty}\sum_{r=1}^{s-1}\sum_{v=l}^{s-1} \mathbb{P}\left\{ \mu^* - \overline{X}_r^* \geq c\sqrt{\frac{\ln s}{r}} \right\} \\
& + \mathbb{P}\left\{ \overline{X}_{i,v} - \mu_i \geq c\sqrt{\frac{\ln s}{v}} \right\} \\
& + \mathbb{P}\left\{ \mu^* < \mu_i + 2c\sqrt{\frac{lns}{v}} \right\}
\end{aligned}$$

The first two probabilities can be bounded using Chernoff inequality (Theorem 3.1)

$$\mathbb{P}\left\{ \mu^* - \overline{X}_r^* \geq c\sqrt{\frac{\ln s}{r}} \right\} \leq \exp\left(-2r\left(c\sqrt{\frac{\ln s}{r}} \right)^2 \right) = s^{-2c^2}$$

And similarly $\mathbb{P}\left\{ \overline{X}_{i,v} - \mu_i \geq c\sqrt{\frac{\ln s}{v}} \right\} \leq s^{-2c^2}$. This turns our bound in

$$\mathbb{E}[T_i(t)] \leq l + \sum_{s=1}^{\infty}\sum_{r=1}^{s-1}\sum_{v=l}^{s-1} 2s^{-2c^2} + \mathbb{P}\left\{ \mu^* < \mu_i + 2c\sqrt{\frac{lns}{v}} \right\}$$

The event $\mu^* < \mu_i + 2c\sqrt{\frac{lns}{v}}$ is true only when $v < 4c^2\frac{\ln(s)}{(\mu^*-\mu_i)^2}$, thus, taking $\hat{l} = \left\lceil 4c^2\frac{\ln(t)}{(\mu^*-\mu_i)^2} \right\rceil$ we have

$$\begin{aligned}\mathbb{E}\{T_i(t)\} \leq & \hat{l} + \sum_{s=1}^{\infty}\sum_{r=1}^{s}\sum_{v=\hat{l}}^{s} 2s^{-2c^2} \\ \leq & 4c^2\frac{\ln(t)}{(\mu^* - \mu_i)^2} + 1 + 2\sum_{s=1}^{\infty} s^{-2c^2+2} \\ = & 4c^2\frac{\ln(t)}{(\mu_i - \mu^*)^2} + 1 + 2\varphi(2c^2 - 2)\end{aligned}$$

This gives the desired bound □

Keeping the notation of the proof of Theorem 3.2, the regret at stage t is given by

$$t\mu^* - \sum_{s=0}^{t} x_s^{I_s} = t\mu^* - \sum_{i=1}^{K}\sum_{I_s=i} x_s^i$$

Applying the expected value operator and summing over the sets of arms we have

$$\begin{aligned}\mathbb{E}\left[t\mu^* - \sum_{s=0}^{t} x_s^{I_s}\right] = & t\mu^* - \mathbb{E}\left[\sum_{i=1}^{K}\sum_{I_s=i} x_s^i\right] \\ = & t\mu^* - \sum_{i=1}^{K}\mathbb{E}\left[T_i(t)\right]\mathbb{E}\left[x_s^i\right] \\ = & t\mu^* - \sum_{i=1}^{K}\mathbb{E}\left[T_i(t)\right]\mu_i\end{aligned}$$

Note that for a fixed time index t the variables $T_i(t)$ satisfy the linear relation $t = \sum_{i=1}^{K} T_i(t)$ hence

$$\begin{aligned}
\mathbb{E}\left[t\mu^* - \sum_{s=0}^{t} x_s^{I_s}\right] =& t\mu^* - \sum_{i=1}^{K} \mathbb{E}\left[T_i(t)\right]\mu_i \\
&= \sum_{i=1}^{K}(\mu^* - \mu_i)\mathbb{E}\left[T_i(t)\right] \\
&= \sum_{i\neq i^*}(\mu^* - \mu_i)\mathbb{E}\left[T_i(t)\right] \\
&\leq \sum_{i\neq i^*}(\mu^* - \mu_i)\left(4c^2\frac{\ln(t)}{(\mu_i - \mu^*)^2} + 1 + 2\varphi(2c^2-2)\right) \\
&= 4c^2\ln(t)\sum_{i\neq i^*}\frac{1}{\mu^* - \mu_i} + \big(1 + 2\varphi(2c^2-2)\big)\sum_{i\neq i^*}(\mu^* - \mu_i)
\end{aligned}$$

In [12], the constant c is set at $\sqrt{2}$ which turns the previous bound in

$$8\ln(t)\sum_{i\neq i^*}\frac{1}{\mu^* - \mu_i} + (1 + 2\varphi(2))\sum_{i\neq i^*}(\mu^* - \mu_i) = 8\ln(t)\sum_{i\neq i^*}\frac{1}{\mu^* - \mu_i} + \left(1 + \frac{\pi^2}{3}\right)\sum_{i\neq i^*}(\mu^* - \mu_i)$$

Since $\varphi(n) \leq \varphi(m)$ for $n > m$, our modification of the bound shows that taking bigger values of c is better. Since a large value of c promotes exploration, our bound shows that when implementing Upper Confidence Bound policy, exploring more is better.

It is to be noted that the resulting upper bound on the regret increases with the number of stages. This might seem counter-intuitive for one would expect that a good decision rule for multi-armed bandits reduces the regret. Nonetheless it can be proved that for any fixed decision rule the expected regret will increase with the number of stages, moreover it will increase at least logarithmically over time, as proved in Theorem 1 of [14]. Thus, the Theorem 3.2 shows that Upper Confidence Bound policy achieves the lowest growth rate possible.

3.2 Combinatorial multi-armed bandits and Combinatorial Upper Confidence Bound

In a regular multi-armed bandit a single arm has to be pulled at each process' stage. In the combinatorial and more general version, a subset of the set of arms (known as super-arm) can be pulled at each stage with some constraints. This makes the problem harder since the size of the exploration set increases and thus it requires larger samples for accurate estimations.

In order to shade some light on the need of this generalization as well as over its under-

standing, let's consider an example of a combinatorial optimization problem that involves randomness. First, we will describe the problem when the underlying probability distributions are known and afterwards we will show how the concept of Combinatorial Multi-Armed Bandit (CMAB) is very helpful when the probabilities are unknown but the problem still needs to be solved. Modeling the combinatorial optimization problem with unknown parameters as a CMAB lets us accurately estimate the parameters of the problem while simultaneously obtaining a solution with a low investment in the sampling resources.

3.2.1 Probabilistic Maximum Coverage Problem

The aforementioned problem is known as the Probabilistic Maximum Coverage Problem (PMCP). We will described it as introduced in [15]. In this instance we are given a set of customers $E = \{1, \cdots, n\}$ and a collection of advertisements $\{C^1, \cdots, C^M\}$. Each customer i has a probability μ_{ij} of clicking on the advertisement C^j. Due to budget constraints, only B of those advertisements will be published and thus the task is to pick the subset of the collection of size B that maximizes the expected number of clicks.

3.2.1.1 The combinatorial optimization problem Consider a fixed choice of B distinct advertisements of the collection $\{C^1, \cdots, C^M\}$, say, $S = \{C^{j_1}, \cdots, C^{j_B}\}$. We say that customer i is covered by the collection S if $\mu_{ij} > 0$ for some j in the indexes of the choice S ($j \in \{j_1, \cdots, j_B\}$).

For the customer $i \in E$, the probability of non-clicking in non of the advertisements of S is $(1 - \mu_{ij_1}) \cdot (1 - \mu_{ij_2}) \cdots (1 - \mu_{ij_B})$. Thus, the probability that the customer i clicks on one of the advertisements of S is

$$1 - \prod_{l=1}^{B} (1 - \mu_{ij_l})$$

Therefore the expected number of clicks of a collection of B advertisements S is

$$\sum_{i=1}^{n} 1 - \prod_{C^j \in S} (1 - \mu_{ij})$$

Since the expected number of clicks depends only on the indexes of the collection of advertisements and the success probabilities μ_{ij}, from now on we will denote our choices of advertisements with set of indexes instead of set of advertisements $S = \{j_1, \cdots, j_B\}$ and the corresponding expected number of clicks will be

$$r_\mu(S) := \sum_{i=1}^{n} 1 - \prod_{j \in S}(1 - \mu_{ij})$$

Under this notation, our problem is to choose the B columns of the matrix $\mu = (\mu_{ij}) \in \mathbb{R}^{n \times M}$ maximizing $r_\mu(S)$ subject to the constraint $|S| = B$.

$$\max_{S \subseteq \{1, \cdots, B\}} r_\mu(S) \tag{13}$$

$$\text{s.t.:} \tag{14}$$

$$|S| = B \tag{15}$$

Unfortunately, solving large instances of problem 13 is computationally unattainable. In did, problem 13 is NP-complete (see [16] section 8.3.2). So, in order to solve large instances of the problem so as to satisfy practical requirements, we will implement an heuristic. But prior to introduce the approximation algorithms concisely we need to verify some properties of the function $r_\mu(S)$ that are crucial to quantify how well the heuristic works

Lemma 3.3. *Given the sets of indexes $R \subseteq S \subseteq \{1, \cdots, M\}$ and the element $k \notin S$, the following inequalities hold*

1.

$$r_\mu(S) \leq r_\mu(S \cup \{k\})$$

2.

$$r_\mu(S \cup \{k\}) - r_\mu(S) \leq r_\mu(R \cup \{k\}) - r_\mu(R)$$

Proof.

1. Since $0 < 1 - \mu_{ik} < 1$ for every i then

$$r_\mu(S \cup \{k\}) = \sum_{i=1}^{n} 1 - (1 - \mu_{ik}) \prod_{j \in S}(1 - \mu_{ij}) \geq \sum_{i=1}^{n} 1 - \prod_{j \in S}(1 - \mu_{ij}) = r_\mu(S)$$

2. The difference $r_\mu(S \cup \{k\}) - r_\mu(S)$ is equal to

$$\sum_{i=1}^{n}(1 - \mu_{ik}) \prod_{j \in S}(1 - \mu_{ij}) - \sum_{i=1}^{n} \prod_{j \in S}(1 - \mu_{ij}) = \sum_{i=1}^{n} \underset{j \in S_{ik}}{\mu} \prod(1 - \mu_{ij})$$

$$\begin{aligned} r_\mu(S \cup \{k\}) - r_\mu(S) &= \sum_{i=1}^{n} \underset{j \in S_{ik}}{\mu} \prod (1 - \mu_{ij}) \\ &= \sum_{i=1}^{n} \mu_{ik} \left(\prod_{j \in S \setminus R} (1 - \mu_{ij}) \right) \left(\prod_{j \in R} (1 - \mu_{ij}) \right) \\ &\leq \sum_{i=1}^{n} \mu_{ik} \prod_{j \in R} (1 - \mu_{ij}) \\ &= r_\mu(R \cup \{k\}) - r_\mu(R) \end{aligned}$$

□

When a set function verifies the first property of Lemma 3.3 it is said to be non-decreasing. As for the second property, it is called sub-modularity [17] . Put in to words a function is sub-modular non-decreasing when its growth rate diminish with the size of a set and thus they are used to generalize the concept of concavity to set functions. Bellow we describe an algorithm specialized for the approximation of the PMCP but that serves to all the combinatorial maximization problems when the objective function is sub-modular.

Algorithm 1 Greedy algorithm for approximation of the PMCP

Require: $\mu_{ij} \quad 1 \leq i \leq n,\ 1 \leq j \leq M,\ B \in \mathbb{N}$
Ensure: A nearly optimal choice of B advertisements S.

```
S ← argmax_{j∈{1,···,B}} r_μ({j})
for each 1 < k ≤ B do
    S ← argmax_{j∈S\{1,···,B}} r_μ(S ∪ {j})
end for
return S.
```

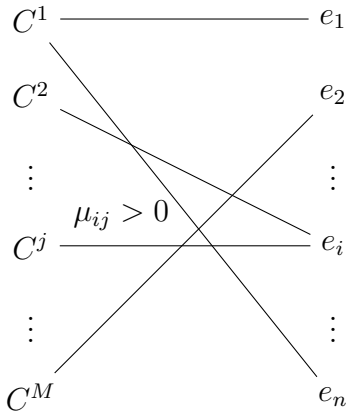
Theorem 4.1 of [17] proves that implementing 1 gives, in the worst case scenario, a choice S such that

$$\frac{r_\mu(S)}{r_\mu(S^*)} \geq 1 - \frac{1}{e}$$

where S^* is the actual optimal solution to 13

3.2.1.2 The combinatorial optimization problem as a CMAB Of course, when the exact values of the μ_{ij} are not known, neither the exact problem nor an heuristic can be

solved. This is why it is useful to model the problem as a MAB because one would like to estimate the unknown parameters in order to solve the problem approximately to a minimum level of accuracy investing the least sampling resources. In this example, the arms will be a set of Bernoulli variables indexed by the amount of relations i, j where $\mu_{ij} > 0$ and the feasible set of arms to pull at each stage (the super-arms) will be the set of those variables induced by some choice of B advertisements. To better understand the arms and super-arms in this example consider the following bipartite graph



Using this visual aid, our arms will be the edges of the graph and the super-arms will be the sets of those edges resulting from picking B nodes from the left side and selecting all the edges starting at the picked nodes. As in the regular version of MAB it would be desirable to estimate the unknown parameters and provided the estimations are good enough, make optimal decisions. To that effect we implement Combinatorial Upper-Confidence Bound. Just as in the non-combinatorial case, a record is kept of the number of times a particular arm has been picked and its average response i.e. $T_{(i,j)}(t)$ and $\overline{X}_{(i,j),t}$. Afterwards, for each combination of customer i and advertisment C^j, the clicking probability μ_{ij} is estimated in a way that arms that have been explored few or have performed well so far, have more weight

$$\overline{\mu}_{ij}(t) = \overline{X}_{(i,j),T_{(i,j)(t-1)}} + c\sqrt{\frac{\ln(t)}{T_{(i,j)(t-1)}}}$$

This estimation is used as input in the Algorithm 1. Subsequently the resulting optimal choice of advertisements is published and based on the observed results the variables $T_{(i,j)}(t)$ and $\overline{X}_{(i,j),t}$ are updated.

Algorithm 2 is stated in terms of the PMCP though it serves a wide variety of CMAB problems. Note that in the optimization step we used the heuristic 1 instead of solving 13. In a general framework of CMAB, this step is called the *approximation oracle*. An $(\alpha, \beta)-$approximation oracle is an algorithm that takes the estimation μ an outputs a super-

Algorithm 2 Combinatorial Upper Confidence Bound (CUCB)

1: For each arm-index (i,j) keep record of $T_{(i,j)}(t)$ and $\overline{X}_{(i,j),t}$ at every stage t.
2: **while** $t \leq T$ **do**
3: $t \leftarrow t+1$
4: For each arm (i,j) set $\overline{\mu}_{i,j} = \overline{X}_{(i,j),T_{(i,j)}(t-1)} + \sqrt{\frac{3\ln(t)}{2T_{(i,j)}(t)}}$.
5: Using the $\overline{\mu}_{i,j}$ as input probabilities , solve an instance of Algorithm 1.
6: Pick the resulting optimal response and update $T_{(i,j)}(t)$ and $\overline{X}_{(i,j),t}$ based on observations.
7: **end while**

arm S such that

$$\mathbb{P}\left(r_\mu(S) \geq \alpha \operatorname{Opt}_\mu\right) \geq \beta$$

In the PMCP Opt_μ is the optimal value of problem 13 using $\mu_{i,j}$ as input probabilities. Lemma 3.3 implies that performing Algorithm 1 is in fact a $\left(1-\frac{1}{e},1\right)-$oracle.

For a fixed arm i,j let $\mathcal{S}_{i,j}$ be the set of super arms containing (i,j) such that the performance is worst than the one expected from the $(\alpha,\beta)-$oracle i.e.

$$\mathcal{S}_{i,j} := \{S : r_\mu(S) < \alpha \operatorname{Opt}_\mu \ S_{ij} = 1\}$$

In [15] they call $\mathcal{S}_{i,j}$ the set of bad super-arms containing the arm (i,j). In the case of the PMCP this would be

$$\mathcal{S}_{i,j} := \left\{S : r_\mu(S) < \left(1-\frac{1}{e}\right) \operatorname{Opt}_\mu \ S_{ij} = 1\right\}$$

Finally, define

$$\Delta_{min}^{i,j} = \left(1-\frac{1}{e}\right) \operatorname{Opt}_\mu - \max\{r_\mu(S) : S \in \mathcal{S}_{i,j}\}$$

$$\Delta_{max}^{i,j} = \left(1-\frac{1}{e}\right) \operatorname{Opt}_\mu - \min\{r_\mu(S) : S \in \mathcal{S}_{i,j}\}$$

and

$$\Delta_{min} = \min_{i,j} \Delta_{min}^{i,j}$$

$$\Delta_{max} = \max_{i,j} \Delta_{max}^{i,j}$$

Theorem 3.4. *Assume the reward function r_μ satisfies all the following hypothesis*

1. ***Monotonicity:*** *If $\mu_1 \geq \mu_2$ (entry-wise) then $r_{\mu_1}(S) \geq r_{\mu_2}(S)$, for every super-arm S.*

2. ***Bounded smoothness:*** *There is a function f that is continuous and strictly increasing such that $f(0) = 0$ and*

$$\max_{i \in S} |(\mu_1)_i - (\mu_2)_i| < \Delta \Rightarrow |r_{\mu_1}(S) - r_{\mu_2}(S)| \leq f(\Delta)$$

for every S, μ_1, μ_2 and every $\Delta > 0$.

Then, the regret of the CUCB Algorithm using an α, β approximation oracle up to stage t is less or equal than

$$n\Delta_{max}\left(\frac{6\ln(t)}{f^{-1}(\Delta_{min})^2} + \frac{\pi^2}{3} + 1\right)$$

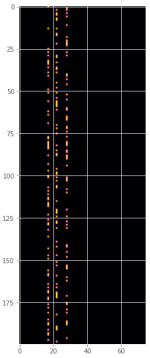
Proof. The proof can be found at [15] (Theorem 1). □

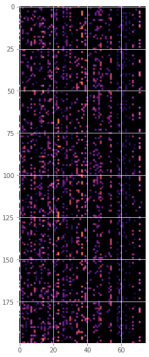
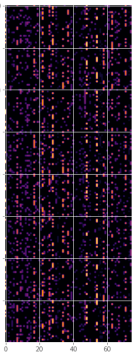
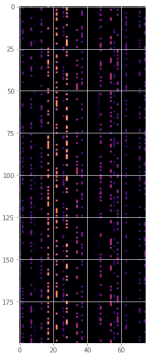
It can be proved that the reward function for PMCP satisfies both, the monotonicity property and the bounded smoothness (see Section 4.1 of [15]). Thus, we have a asymptotically optimal policy that suggests solutions sequentially while estimating the actual parameters of the problem. To conclude this chapter, some computational demonstrations of the CUCB applied to the PMCP will be given.

3.2.2 Computational examples

In the following example, a set of $n = 200$ customers and a collection of $M = 75$ advertisements was generated randomly as well as the actual clicking probabilities $\mu_{i,j}$. Given a budget of $B = 3$ advertisements the space of super-arms has size $\binom{M}{B} = 67525$ but only $T = 1000$ sample stages were allowed. In this computational demonstration the optimal solution and the current super-arm will be displayed as a sparsity plot in which the B selected columns (advertisements) are highlighted. In order to visualize the effectiveness of CUCB, together with the sparsity plot we show a heat map showing the frequency at which each arm (i, j) has been selected. The pixels (corresponding to the entries of a $n \times M$ matrix i.e. the arms) colored with brighter colors have been selected more frequently. In other words, the frequency heat plots are heat maps of the matrix with values $T_{i,j}$.

Actual optimal super-arm



$T_{i,j}$ values at stage $t = 50$	$T_{i,j}$ values at stage $t = 100$	$T_{i,j}$ values at stage $t = 1000$
		

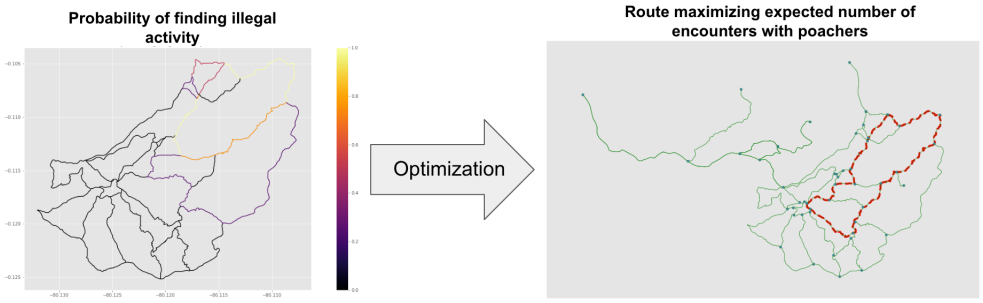
As the number of stages increases, the heat tends to concentrate in the B optimal columns of the matrix, depicting the asymptotic behavior of CUCB. A larger sequence of images containing the heat maps and super-arms and more relevant visualizations for every stage, can be found in this link.

4 CMAB for suggesting patroling routes

The ranger's problem is to choose columns of C_G (the matrix of cycles of a graph G) sequentially considering patrolling constrains so that the success of the patrolling policy is maximized. The only method for solving it considered so far is computationally unattainable for cases of practical interest and makes unrealistic assumptions on the poacher. In this section we will model the ranger's problem as a stochastic combinatorial optimization problem. Acknowledging that some of the input parameters of the stochastic combinatorial optimization problem are unknown, the problem will be modeled as a CMAB. The advantages of this approach against the Stackelberg approach include the consideration of additional relevant variables such as the changing availability of resources at each patrolling stage and the computational efficiency. In fact, we will provide computational examples over the actual graph of trails of Jama Coaque Ecological Reserve.

As for the assumptions made for modeling the ranger's problem this way, we suppose that there is a collection of Bernoulli independent variables indexed by the trails of the reserve, each modeling the event of finding illegal activity on the trail. We will also assume that this distributions remain the same at every patrolling stage. Provided the success probabilities are known, every available ranger should pick the circuit-like route that that maximizes the expected number of encounters with the poacher.

The heat map of success probabilities on each trail determines a unique optimal route



4.1 The combinatorial optimization problem

Let B be the number of available rangers at the reserve, each willing to choose a feasible route. Introducing this new parameter, the problem of the rangers can be described as choosing the B columns of C_G which maximize the covered terrain taking into account the probability of finding a poacher. Based on this notation, the number of covered trails is given by

$$U_r(t,\eta) = \sum_{j=1}^{M} \eta_j \sum_{i=1}^{n} t_i C_i^j = t^T C_g \eta$$

where $\eta \in \{0,1\}^M$ is the choice of the team of rangers

$$\eta_j = \begin{cases} 1 & \text{Some ranger picked circuit } j \\ 0 & \text{Otherwise} \end{cases}.$$

Therefore, if μ_i is the probability of finding illegal activity in the $i-$th trail, the expected number of encounters with illegal activity given the choice η is

$$r_\mu(\eta) := \mathbb{E}[U_r(t,\eta)] = \sum_{j=1}^{M} \eta_j \sum_{i=1}^{n} \mu_i C_i^j$$

A penalty of ε could be charged for those visited trails on which non illegal activity was found, resulting in a regularization of the choice η that maximizes the expected number of encounters with illegal activity

$$r_\mu(\eta) = \sum_{j=1}^{M} \eta_j \sum_{i=1}^{n} \mu_i C_i^j - \varepsilon C_i^j (1-\mu_i)$$

The reason why the penalization is not made over the unguarded trails by the rangers in which there was illegal activity (as in the game theoretical case), is that it doesn't make sense to assume that the rangers have information about the trails no one visited. Besides, this choice of reward functions captures the fact that when committing resources to some trails we are paying the price of not obtaining information for some areas within the natural reserve. For the rest of the document we will set $\varepsilon = 0$

Theorem 4.1 (M. Velasco, N. Betancourt)**.** *If the team of rangers knows the values of the vector of probabilities μ, then the choice η that maximizes $r_\mu(\eta)$ is a optimal solution to*

$$(RP)_\mu \begin{cases} \max\limits_{\eta \in \mathbb{R}^M, y \in \mathbb{R}^n} & \sum \mu_i y_i \\ & s.t.: \\ & \sum_{j=1}^{M} \eta_j = B \\ & y_i = \sum_{j=1}^{M} C_i^j \eta_j \\ & \eta_j \in \{0,1\}, y_i \in \{0,1,\cdots,B\} \end{cases} \tag{16}$$

Proof. Note that

$$\begin{aligned} r_\mu(\eta) &= \sum_{j=1}^{M} \eta_j \left[\sum_{i=1}^{n} \mu_i C_i^j \right] \\ &= \sum_{i=1}^{n} \sum_{j=1}^{M} \eta_j (\mu_i C_i^j) \\ &= \sum_{i=1}^{n} \mu_i \left[\sum_{j=1}^{M} C_i^j \eta_j \right] \\ &= \sum_{i=1}^{n} \mu_i (C_G \eta)_i \end{aligned}$$

for every η. Then, in the specific case of an optimal solution y^*, η^* to 16

$$y_i^* = \sum_{j=1}^{m} C_i^j \eta_j^* = (C_G \eta)_i \Rightarrow \sum_{i=1}^{n} \mu_i y_i^* = \sum_{i=1}^{n} \mu_i (C_G \eta)_i$$

therefore, the optimal value of $(RP)_\mu$ is equal to $r_\mu(\eta^*)$. Since the previous reasoning is true for any η, r_μ is upper bounded by $r_\mu(\eta^*)$. □

The formulation of problem 16 might be similar to that of the PMCP for we are trying to cover the set of trails using circuits and considering the poaching probabilities μ_i. Nonetheless, these two problems are very different. The main difference between them is that random variables in the ranger's problem are indexed by the trails and not by circuit-trail pairs as in PMCP. This small difference in the problem formulation translates into a big computational difference. The resemblance of Problem 16 with a PMCP suggests the implementation of heuristic such as 1 for finding approximate solutions efficiently. However, 16 has a computational advantage over PMCP since the relaxation of problem 16 is in fact equivalent to the original MILP, meaning that we have a $(1,1)-$approximation oracle which is computationally less intensive than the original MILP.

Theorem 4.2 (M. Velasco, N. Betancourt)**.** *Consider the relaxation of* $(RP)_\mu$ *given by*

$$(\widetilde{RP})_\mu \begin{cases} \max\limits_{\eta \in \mathbb{R}^M, y \in \mathbb{R}^n} & \sum \mu_i y_i \\ & s.t.: \\ & \sum_{j=1}^{M} \eta_j = B \\ & y_i = \sum_{j=1}^{M} C_i^j \eta_j \\ & \eta_j \in [0,1], y_i \in [0,B] \end{cases} \tag{17}$$

and let $\tilde{\eta}, \tilde{y}$ be optimal solutions. If η is the choice of B columns of C_G resulting from an iterated sampling of the discrete distribution $\tilde{\eta}/B$ (independently and with replacement) then the expected value of $r_\mu(S)$ is equal to the optimal value of 17

Proof. Let Z_i be the random variable that counts the number of cycles in η crossing the $i-$th trail. For $1 \leq l \leq B$ let X_l^i the random variable which is 1 when the $l-$th sampled cycle contains the $i-$th trail. Then $Z_i = \sum_{l=1}^{B} X_l^i$. Since the sample is independent and with replacement, all of the X_l^i are i.i.d Bernoulli random variables with success probabilities given by

$$\sum_{j:i\in C^j} \tilde{\eta}_j/B = \sum_{j=1}^{M} C_i^j \tilde{\eta}_j/B = y_i/B$$

This proves that the expected value of Z_i is $\tilde{y}_i$. Note that $r_\mu(S) = \sum \mu_i Z_i$ and then the expected value of $r_\mu(S)$ is $\sum \mu_i E[Z_i] = \text{Opt}(\widetilde{RP})_\mu$ where $\text{Opt}\,(\widetilde{RP})_\mu$ the optimal value of 17. □

4.2 The Ranger's problem as a CMAB

In practice rangers do not know the probabilities of illegal activities along the various trails and very few historical poaching data might be available. As done in section 3 with the Probabilistic Maximum Coverage Problem, we overcome the absence of essential input information by modeling the Ranger's combinatorial optimization problem as a CMAB. An additional difference between PMCP and the Ranger's problem is that when modeled as a CMAB, the arms in PMCP are indexed by the set-element relations i, j while in the ranger's problem the arms will be indexed only by the elements i a.k.a. the trails.

More precisely, in this CMAB instance the super-arm choice will be modeled by vectors $\eta \in \{0,1\}^M$ with exactly B non-null entries and the arms are the Bernoulli random variables indexed by the trail. We will implement the general CUCB policy to the ranger's problem described in algorithm 2. In this context $T_i(t)$ is the number of times that the $i-$th trail has been patrolled up to stage t and $\overline{X}_{i,t}$ is the percentage of those times on which illegal activity has been detected. Besides understanding the CUCB variables in the context of the ranger problem, in this section we will provide computational demonstration of the method and point out that the combinatorial structure of the set of super-arm can be exploited in order to reduce the stages devoted to exploration.

In the first place, in order to have theoretical guarantees on the asymptotic optimallity of the regret, we need to check the hypothesis of Theorem 3.4 over the reward function $r_\mu(\eta)$.

Lemma 4.3. *Fix a choice η of B cycles*

1. ***Monotonicity:*** *If $\mu_1 \geq \mu_2$ (entry-wise) then $r_{\mu_1}(\eta) \geq r_{\mu_2}(\eta)$.*

2. ***Bounded smoothness:*** *There is a function f that is continuous and strictly increasing such that $f(0) = 0$ and*

$$\max_{i \in \eta} |(\mu_1)_i - (\mu_2)_i| < \Delta \Rightarrow |r_{\mu_1}(\eta) - r_{\mu_2}(\eta)| \leq f(\Delta)$$

for every μ_1, μ_2 and every $\Delta > 0$.

Proof.

1. From the proof of Theorem 4.1 $r_\mu(S)$ could be written as

$$\sum_{i=1}^{n} \mu_i (C_G \eta)_i$$

which implies that the reward function is monotonically increasing on μ when η is fixed.

2. Let μ^1, μ^2 and $\Delta > 0$ be variables such that $\max_{i \in \eta} |(\mu_1)_i - (\mu_2)_i| < \Delta$. Observe that

$$r_\mu(\eta) = \sum_{i=1}^{n} \mu_i (C_G \eta)_i$$

and therefore

$$\begin{aligned} |r^1_\mu(\eta) - r^2_\mu(\eta)| =& |\sum_{i=1}^{n} (\mu^1_i - \mu^2_i)(C_G \eta)_i| \\ &\leq \Delta \sum_{i=1}^{n} (C_G \eta)_i \\ &\leq nB\Delta \end{aligned}$$

Thus, it suffices to take $f(x) = nBx$ to finish the proof.

□

The previous lemma implies that if for the $i-$th trail the probability of finding illegal activity μ_i increases in x while keeping all other probabilities constant, the number of encounters with the poacher will increase to at most xnB.

In the following pseudo-code, we re-write Algorithm 2 for the context of the Ranger's Problem

Algorithm 3 CUCB for the Ranger's Problem

1: For each trail i keep record of $T_i(t)$ and $\overline{X}_{i,t}$ at every stage t.
2: **while** $t \leq T$ **do**
3: $t \leftarrow t+1$
4: For each trail i set $\mu_i = \overline{X}_{i,T_i(t-1)} + \sqrt{\frac{3\ln(t)}{2T_{(i}(t)}}$.
5: Using the μ_i as input probabilities , solve an instance of 17 .
6: Patroll the resulting optimal route and update $T_i(t)$ and $\overline{X}_{i,t}$ based on observations.
7: **end while**

It is worth noting that in [15] they suggest an initialization routine of CUCB in which for each arm one has to pull a super-arm containing that arm. This routine previous to the sequential optimization step might be costly in some cases but in our case less stages has to be invested to exploration due to the combinatorial structure of our super-arms. For instance, the optimal solution to the following LP indicates a minimal collection of cycles that crosses each trail at least once

$$\begin{aligned} &\min_{\eta\in\mathbb{R}^M} \sum \eta_j \\ &\text{s.t.:} \\ &\qquad \sum_{j=1}^{M} C_i^j \eta_j \geq 1 \\ &\qquad \eta_j \in \{0,1\} \end{aligned}$$

For the computational examples in the next section, we used this optimal initialization routine reducing the investment of $n = 138$ initialization stages to 6.

4.2.1 CUCB regret bound for the Ranger's Problem

The Lemma 4.3 shows that implementing algorithm 0 has the same asymptotic regret stated in Theorem 3.4. Bellow we will analyze this bound when understood in the context of the Ranger's Problem

Let Opt_μ be $\text{Opt}(\widetilde{RP})_\mu$, the optimal value of problem 17 when μ is used as input probability vector of finding illegal activity on the trails. Recall from section 3 that for a trail i the set of bad super arms $\mathcal{S}_i$ is the collection of subset of size B of of cycels traversing the ith trail

$$\mathcal{S}_i := \{S : r_\mu(\eta) < \text{Opt}_\mu \ \ \eta_i = 1\}$$

In the context of the Ranger's Problem, the quantities

$$\Delta^i_{max} = \text{Opt}_\mu - \min\{r_\mu(\eta) : \eta \in \mathcal{S}_i\}$$

$$\Delta^i_{min} = \text{Opt}_\mu - \max\{r_\mu(\eta) : \eta \in \mathcal{S}_i\}$$

are the difference between the performance of the optimal route with the worst and the second best route respectively and the variables

$$\Delta_{min} = \min_i \Delta^{i,j}_{min}$$

$$\Delta_{max} = \max_i \Delta^{i,j}_{max}$$

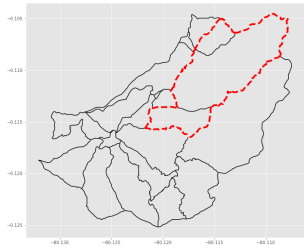
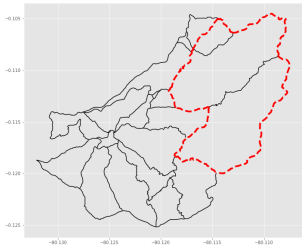
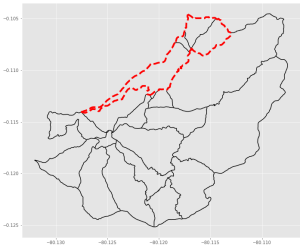
are the minimum and maximum of those quantities over the trails

Since our bounded smoothness function is $f(x) = nBx$, the regret bound of CUCB for the Ranger's Problem depends on the number of trails and the number of available rangers

$$n\Delta_{max}\left(\frac{6\ln(t)}{f^{-1}(\Delta_{min})^2} + \frac{\pi^2}{3} + 1\right) = n\Delta_{max}\left(\frac{6n^2B^2\ln(t)}{\Delta^2_{min}} + \frac{\pi^2}{3} + 1\right)$$

4.3 A practical implementation

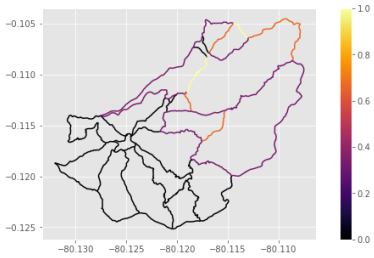
In the next computational demonstration we used the actual map of trails of the Jamacoaque Ecological Reserve in Ecuador that has $n = 138$ trails and $M = 135010$ simple circuits. We simulated a poacher trying to hunt and log within the area of the reserve by following one of three routes starting from a node on the boundary of the map which could be thought is near to her home. The poacher follows those routes at random with a uniform probability

Poacher's random routes		
Episodic poaching route 1	Episodic poaching route 2	Episodic poaching route 3
		

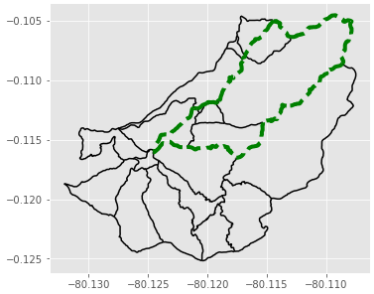
Uniformly visiting those routes induces the following poaching probabilities on each trail

A single ranger ($B = 1$) is watching over the reserve with no historical information and ignoring the trails contained in the routes of the poacher as well as the frequency at which the poacher visits them. In the case the probabilities heat-map were given the actual optimal route would be

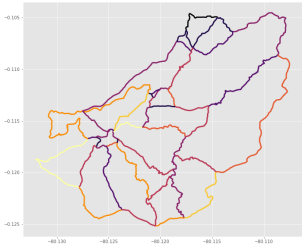
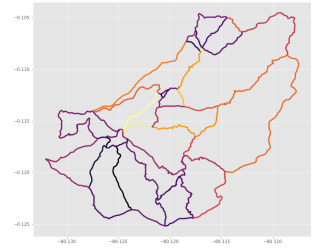
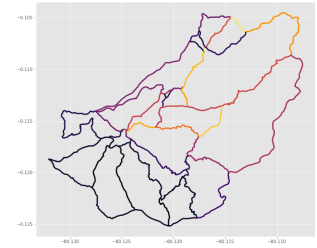
Induced poaching probabilities μ.



Optimal route provided the unknown probabilities.



In the following figures, we see a heat map showing the number of times the ranger has visited every trail at the given stage. Brighter colors corresponding to higher number of visits. As shown bellow, as the number of patrolling stages increases, the heat concentrates on those trails belonging to the actual optimal route even though the ranger ignores every valuable information about the poacher.

Patrolling frequency heat-map induced by CUCB		
Route at stage $t = 10$	Route at stage $t = 50$	Route at stage $t = 180$
		

As in the PMCP computational demonstration, a larger sequence of images containing the heat maps and super-arms and more relevant visualizations for every stage, can be found in this link.

4.4 Future directions

4.4.1 Performance improvement against a reactive poacher

In the previous computational simulations the ranger is able to discover the optimal patrolling route because the synthetic poacher is not modifying her behavior despite the many unwanted encounters with the ranger. Here, our CUCB ranger is opposed with a more sophisticated poacher, but not one that is strategically as sophisticated as the Stackelberg Poacher. This poacher will either change her usual routes trying to avoid past unwelcome meetings with the ranger's route or will stop poaching for a while. Both of these behaviors disrupt the CUCB estimation. In this section we will see how badly CUCB performs when facing a non-stationary scenario, closer to the real one. This bad performance points a necessary future direction in which the sequential route picking takes into account the changing probabilities on the trails in function of the past patrolling routes and its damage to the poacher.

4.4.1.1 Reactive poacher of the first kind: Π_1 This first hypothetical poacher will have a fixed patrolling distribution over some set of routes α and will patrol according to α. But if in stage t the sampled route meets at least one of the trails of the chosen ranger's route, the poacher will skip stage $t+1$ and wait until stage $t+2$ to sample α again. This behavior makes it harder for the ranger to estimate α because the frequency at which he stumbles upon the poacher on those trails belonging to routes on the support of α might seem much lower than the actual probability.

A similar simulation to the one in the non-reactive poacher scenario can be found in this link. This time the ranger will face a Π_1 poacher that hides whenever he gets caught. Note that in some patrolling stages there is no route corresponding to the poacher.

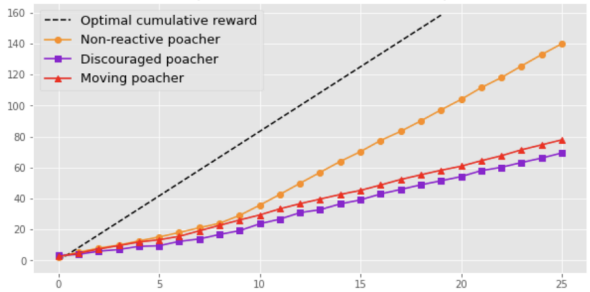
4.4.1.2 Reactive poacher of the second kind: Π_2 A Π_2 poacher is a poacher that has two patrolling distributions α and β over (probably disjoint) sets of routes. The distribution α is preferred over β (because proximity to her home, animal or timber abundance, quality of the trails...) but whenever she gets caught she will switch to distribution β until getting caught again. Note that the resulting patrolling frequency is not a mix of the distributions α and β because whether the poacher picks α or β depends on the ranger's choices and not on a third distribution.

This simulation is similar to the one in the non-reactive scenario but this time the ranger will face a Π_2 poacher that switches her habits temporarily whenever he gets caught. Each of the two bottom rows on the video display the support of the distributions α and β.

4.4.1.3 Reward comparison between the different kind of poachers

In order to depict how bad CUCB performs when the stationarity hypothesis does not holds, we simulated 100 patrolling activities of CUCB of 26 patrolling stages each but facing the three different kinds of poachers. Afterwards we computed the sample average of the obtained reward at each stage and plotted it as a function of the stage number. The black straight line is the optimal reward possible of the stationary scenario i.e. the reward the ranger would have earned provided she had known the probabilities of the non-reactive poacher and would have implemented the optimal route consistently. It is simply a straight line crossing 0 and with slope equal to the optimal value of problem 16.

The cumulative reward during 25 patrolling stages for each kind of ranger.



As one would expect, the most sophisticated the poacher is, the worst performance CUCB has. This points a future research direction that takes into account the unknown way in which the illegal agents could react when caught. This is important since this kind of behaviors are the ones to be expected on the field .

4.4.2 Interaction with additional information

The recent efforts from different actors in science and engineering for enhancing conservation initiatives is providing ranger's with technological tools that give all kind of information regarding illegal activity in protected areas. Remote sensing ([18],[19]) and computer vision specialized to make automatic identification of fauna on the field ([20],[21],[22],[23]) are the most common examples of this. A next interesting research direction would be an smart way of coupling the route suggestion system with additional information provided by satellite imagery, camera traps or other sources. As an example, bellow we suggest a shallow description of how the route suggestion system described in this section could be coupled with an acoustic monitoring system creating a mutual beneficial interaction. Bellow, we give a particular example of how the route suggestion tool could be coupled with an acoustic monitoring system.

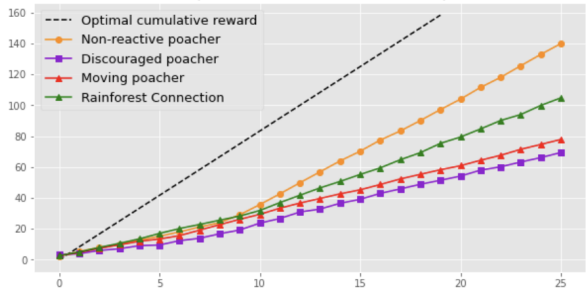
Acoustic identification is proving to be very effective in natural reserves management

when it comes to prevent logging and hunting. Initiatives such as Rainforest Connection and its Guardian Devices are providing rangers in protected areas with real time alerts of illegal activity going on within the area of the protected area. This system can detect sounds up to 1500 meters far from the tree in which the device is installed, covering huge areas but lacking of precision at the time of notifying rangers about the exact spot in which the illegal activity is happening

This kind of systems might be benefited from working simultaneously with the route-suggestion algorithm. If both systems are coupled, besides real time alerts of logging and hunting noises near some point of the reserve, the rangers will be provided with a data-driven suggestion of a route with which they can head towards the area of the thread. Inspired in this possible advantage, we think that future research efforts should be focused on how to properly couple these systems requesting the CUCB algorithm, for instance, that the suggested route in a stage in which a acoustic thread have been detected crosses some trails within the threatened area.

In the other hand, it will also be interesting to explore how the route suggesting algorithm might improve when geard with an acoustic monitoring system. In order to depict this desirable improvement we faced the CUCB algorithm against a reactive poacher again but notifying the ranger every so often that illegal activity is going on in a collection of trails one of which is included in the current route of the poacher. Afterwards, the algorithm solves the problem 17 with the additional constraint that at least on of the trails in the notified collection should be crossed by the optimal route. As shown in this new performance plot, the green line outperforms the lines corresponding to CUCB against the strategical poachers alone. This synthetic simulation might encourage future empirical research on the topic.

Performance improvement when CUCB is coupled with an Acoustic Monitoring System.



References

[1] M. R. Appleton, A. Courtiol, L. Emerton, *et al.*, "Protected area personnel and ranger numbers are insufficient to deliver global expectations," *Nature Sustainability*, 2022.

[2] D. Kar, T. H. Nguyen, F. Fang, M. Brown, A. Sinha, M. Tambe, and A. X. Jiang, "Trends and applications in stackelberg security games," *Handbook of Dynamic Game Theory*, pp. 1223–1269, 2018.

[3] L. Xu, E. Bondi, F. Fang, A. Perrault, K. Wang, and M. Tambe, "Dual-mandate patrols: Multi-armed bandits for green security," *CoRR*, vol. abs/2009.06560, 2020. arXiv: 2009.06560.

[4] M. V. N. Betancourt, "Green security games in graphs," *In preparation*, 2022.

[5] P. Bauer, "The circuit polytope: Facets," *Mathematics of Operations Research*, vol. 22, no. 1, pp. 110–145, 1997.

[6] M. Zhu, X. Zheng, Y. Wang, Y. Li, and Q. Liang, "Adaptive portfolio by solving multi-armed bandit via thompson sampling," eng, *arXiv.org*, 2019.

[7] S. S. Villar, J. Bowden, and J. Wason, "Multi-armed bandit models for the optimal design of clinical trials: Benefits and challenges," eng, *Statistical science*, vol. 30, no. 2, pp. 199–215, 2015.

[8] K. Liu and Q. Zhao, "Adaptive shortest-path routing under unknown and stochastically varying link states," eng, *arXiv.org*, 2012.

[9] A. Badanidiyuru, R. Kleinberg, and A. Slivkins, "Bandits with knapsacks," eng, *Journal of the ACM*, vol. 65, no. 3, pp. 1–55, 2018.

[10] F. Trovò, S. Paladino, M. Restelli, and N. Gatti, "Improving multi-armed bandit algorithms in online pricing settings," eng, *International journal of approximate reasoning*, vol. 98, pp. 196–235, 2018.

[11] S. Ontañón, "Combinatorial multi-armed bandits for real-time strategy games," eng, *The Journal of artificial intelligence research*, vol. 58, pp. 665–702, 2017.

[12] P. Auer, N. Cesa-Bianchi, and P. Fischer, *Finite-time analysis of the multiarmed bandit problem - machine learning*, 2002.

[13] S. Boucheron, G. Lugosi, and P. Massart, "18Basic Inequalities," in *Concentration Inequalities: A Nonasymptotic Theory of Independence*, Oxford University Press, 2013, pp. 18–51. eprint: https://academic.oup.com/book/0/chapter/195070651/chapter-pdf/43899181/acprof-9780199535255-chapter-02.pdf.

[14] T. L. Lai, H. Robbins, *et al.*, "Asymptotically efficient adaptive allocation rules," *Advances in applied mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[15] W. Chen, Y. Wang, and Y. Yuan, "Combinatorial multi-armed bandit: General framework and applications," in *Proceedings of the 30th International Conference on Machine Learning*, S. Dasgupta and D. McAllester, Eds., ser. Proceedings of Machine Learning Research, Atlanta, Georgia, USA: PMLR, 2013, pp. 151–159.

[16] A. Benoit, *A Guide to Algorithm Design : Paradigms, Methods, and Complexity Analysis* (Chapman & Hall/CRC applied algorithms and data structures series), eng, 1st edition. Boca Raton, FL: Taylor and Francis, an imprint of CRC Press, 2013 - 2014.

[17] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[18] S.-H. Lee, K.-J. Han, K. Lee, K.-J. Lee, K.-Y. Oh, and M.-J. Lee, "Classification of landscape affected by deforestation using high-resolution remote sensing data and deep-learning techniques," eng, *Remote sensing (Basel, Switzerland)*, vol. 12, no. 20, pp. 3372–, 2020.

[19] Y. E. Shimabukuro, *Spectral Mixture for Remote Sensing Linear Model and Applications* (Springer Remote Sensing/Photogrammetry), eng, 1st ed. 2019. Cham: Springer International Publishing, 2019.

[20] S. Sagawa, P. W. Koh, T. Lee, *et al.*, "Extending the WILDS benchmark for unsupervised adaptation," *CoRR*, vol. abs/2112.05090, 2021. arXiv: 2112.05090.

[21] P. Kulits, J. Wall, A. Bedetti, M. Henley, and S. Beery, "Elephantbook: A semi-automated human-in-the-loop system for elephant re-identification," *CoRR*, vol. abs/2106.15083, 2021. arXiv: 2106.15083.

[22] M. S. Norouzzadeh, D. Morris, S. Beery, N. Joshi, N. Jojic, and J. Clune, "A deep active learning system for species identification and counting in camera trap images," *CoRR*, vol. abs/1910.09716, 2019. arXiv: 1910.09716.

[23] D. Tuia, B. Kellenberger, S. Beery, *et al.*, "Perspectives in machine learning for wildlife conservation," *Nature Communications*, vol. 13, no. 1, 2022.